Performance Evaluation of Multi-Tier Ensemble Classifiers for Phishing Websites

Jemal Abawajy, Gleb Beliakov, Andrei Kelarev
School of Information Technology
Deakin University, 221 Burwood Hwy,
Burwood 3125, Australia
Email:{jemal.abawajy, gleb, kelarev}@deakin.edu.au

John Yearwood
School of Science, Information Technology
and Engineering, University of Ballarat,
P.O. Box 663, Ballarat, Victoria 3353, Australia
Email:j.yearwood@ballarat.edu.au

***Abstract*—This article is devoted to large multi-tier ensemble classifiers generated as ensembles of ensembles and applied to phishing websites. Our new ensemble construction is a special case of the general and productive multi-tier approach well known in information security. Many efficient multi-tier classifiers have been considered in the literature. Our new contribution is in generating new large systems as ensembles of ensembles by linking a top-tier ensemble to another middle-tier ensemble instead of a base classifier so that the top-tier ensemble can generate the whole system. This automatic generation capability includes many large ensemble classifiers in two tiers simultaneously and automatically combines them into one hierarchical unified system so that one ensemble is an integral part of another one. This new construction makes it easy to set up and run such large systems. The present article concentrates on the investigation of performance of these new multi-tier ensembles for the example of detection of phishing websites. We carried out systematic experiments evaluating several essential ensemble techniques as well as more recent approaches and studying their performance as parts of multi-level ensembles with three tiers. The results presented here demonstrate that new three-tier ensemble classifiers performed better than the base classifiers and standard ensembles included in the system. This example of application to the classification of phishing websites shows that the new method of combining diverse ensemble techniques into a unified hierarchical three-tier ensemble can be applied to increase the performance of classifiers in situations where data can be processed on a large computer.**

***Keywords*-phishing websites; ensemble classifiers; multi-tier ensembles; Random Forest**

I. Introduction

Experiments evaluating classifiers applied to particular areas are important, since their outcomes can be used in order to improve the performance of future applications and can contribute to choosing directions of future research. For any given algorithm that produces very good outcomes in certain applications, there always exist examples of data sets in other domains where different algorithms are more effective. This is also confirmed by the so-called "no-free-lunch" theorems, which imply that there does not exist one algorithm, which is best for all problems [45]. The performance of every category of algorithms depends on the dimension of a data set and the number of instances, types of attributes, the nature of functional relations and dependencies among the attributes and other parameters.

We introduce a new unified multi-tier construction of ensemble classifiers combining diverse ensembles into one integrated hierarchical system. This construction is illustrated in Figure 1. More explanations are given in Section II. Figure 2 shows how to aggregate the classifiers at different levels to obtain the multi-tier construction.

Every ensemble classifier at the middle tier of this construction is an integral part of the ensemble classifier at the top tier, and in turn every base classifier at the bottom tier is included as a part of the ensemble classifier of the middle tier, see Section II for more details. Using one ensemble as an integral part of another ensemble makes it easy to set up and run such ensembles, even though they can be very large.

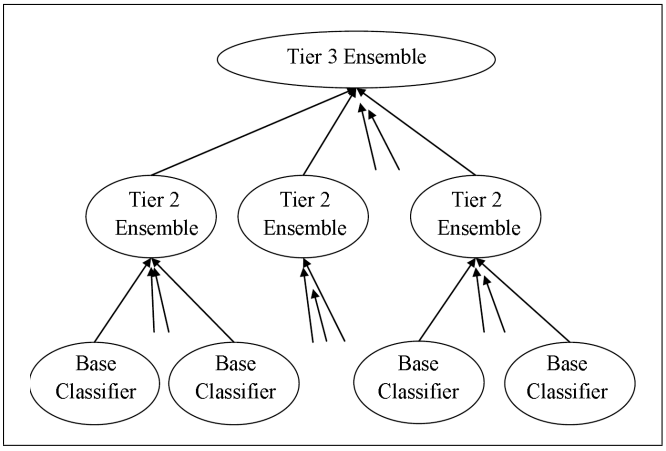

Figure 1. Data flow in three-tier ensemble classifiers generated as ensembles of ensembles by WEKA

The present article is devoted to experiments comparing the performance of new three-tier classifiers, their base classifiers and standard ensemble classifiers in the special case of an application to the detection of phishing websites. While phishing is an important direction that has been actively investigated recently, the aim of our paper is to develop a general technique that may be useful for various applications in information security. Let us refer to the Anti-Phishing Working Group [1], OECD Task Force on

Spam [34] and recent papers [4], [8], [14], [18], [19], [28], [47] for background information and preliminaries on phishing. The authors hope that the outcomes of this example of application prove helpful for the future development of classifiers in other branches of information security too.

Our new results show that novel three-tier ensemble classifiers achieved substantially better performance in comparison with the base classifiers or standard ensemble classifiers. This demonstrates that the new method of combining diverse ensemble techniques into one unified three-tier ensemble incorporating diverse ensembles as parts of other ensembles can be applied to improve classifications.

The paper is organised as follows. Section II describes new multi-tier ensemble classifiers investigated in this paper. Section III is devoted to preprocessing of data. Section IV deals with the base classifiers and ensemble classifiers. Section V contains the outcomes of experiments comparing the effectiveness of base classifiers, ensemble classifiers and three-tier ensemble classifiers. These results are discussed in Section VI. Main conclusions are presented in Section VII.

For consistency, in writing the paper an attempt was made to use present simple tense throughout to describe what is done in this article as well as to refer to background information. Past simple and present perfect tenses were reserved to the discussion of articles published previously and to the description of our experiments, since all our tests had been completed before we started writing the paper.

II. Three-tier Ensemble Classifiers

Ensemble classifiers combine a collection of base classifiers into a common classification system. Here we introduce and explain our new multi-tier ensemble construction inspired by previous research in the literature. Our experiments evaluate performance of such large three-tier ensemble classifiers combining diverse ensemble classifiers on two tiers into one unified system. Several efficient multi-tier classifiers and more general multi-classifier systems have been explored, for example, in the previous publications [19], [20], [23], [24], [25].

Several techniques for the design of ensemble classifiers are well known in artificial intelligence and data mining. This paper introduces a new three-tier construction, which makes it easy to combine diverse ensemble methods into one scheme. Our experiments are devoted to performance evaluation of new large three-tier ensemble classifiers for phishing websites.

This paper deals with large three-tier ensemble classifiers, illustrated in Figure 1. The direction of arrows in the diagram indicates the flow of data. All base classifiers pass their output on to Tier 2 ensemble classifiers. The Tier 2 ensemble classifiers combine the output of base classifiers. Their output in turn in analysed by the Tier 3 ensemble classifier that makes the final decision for the whole multi-tier classification system. Arcs not connected to classifiers indicate the direction of possible data flow from additional classifiers. The whole system may involve thousands of base classifiers, but it is easy to set it up, since in most cases the Tier 2 classifiers generate the whole collection of their base classifiers automatically given just one instance of a base classifier. Likewise, all Tier 2 ensemble classifiers are generated by the Tier 3 ensemble classifier automatically given only one instance of a Tier 2 ensemble classifier. This means that the Tier 3 ensemble classifier generates its Tier 2 classifiers and executes them in exactly the same way as it usually handles base classifiers. Similarly, each Tier 2 ensemble applies its method to combine its base classifiers as usual. The whole system is generated automatically in SimpleCLI, as illustrated in Figure 2.

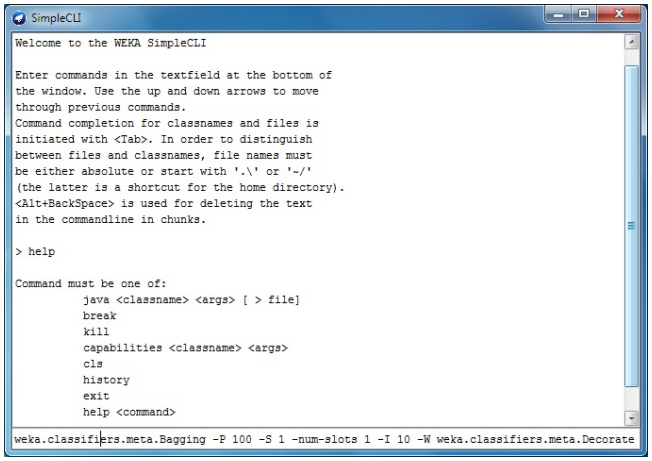

Figure 2. A part of command line generating three-tier ensemble in SimpleCLI

Thus, in this paper we introduce and investigate a three-tier ensemble construction originating as a contribution to the general approach introduced by previous authors. We obtain new results evaluating performance of such large three-tier ensemble classifiers. These new results show, in particular, that Random Forest performed best in this setting for our data set considered in this article, and that novel three-tier ensemble classifiers can be used to achieve further improvement of the classification outcomes. The three-tier ensemble classifiers based on Random Forest achieved better performance compared with the base classifiers or simpler ensemble classifiers.

Large three-tier ensemble classifiers require a lot of computer memory to train, especially for very large data sets, where they can be used to improve performance. If a data set is small and an ensemble classifier is larger, then it will revert to using just one base classifier and produce the same outcomes as the base classifier. As we will see in Section V below, our experiments show that such large three-tier ensemble classifiers are effective if diverse ensembles are combined at different tiers of the three-tier ensemble

classifier. The authors believe that this approach to designing ensembles of classifiers deserves further investigation for other large data sets and application directions too.

III. Feature Extraction

We used the same set of features extracted from the data set of phishing websites considered by the authors in [4], since it is suitable for this study. Our new experiments used a collection of simple features extracted during work on the paper [4]. Similar data sets are available from the downloadable databases at the PhishTank [35]. The present article investigates a novel method for improving performance of the classifiers, and we did not attempt to extract more sophisticated collections of features. The extraction of features is very important for applications, for example, see [2], [21], [22], [26], [29], [30], [31], [33], [40], [41] and [42], but it is not the main focus of the present article.

Since this paper concentrates on the contribution of multi-tier ensembles, for the purposes of this work, we applied the bag-of-words model and extracted only a simple collection of the features reflecting the content of the websites. As in [4], we used *term frequency–inverse document frequency* word weights, or TF-IDF weights, to select words as features. Features were extracted using a flexible preprocessing and feature extraction system implemented in Python by the third author.

We collected a set of words with highest TF-IDF scores in all websites of the data set. For each website, the TF-IDF scores of these words in the website were determined. These weights and additional features were assembled in a vector. In order to determine the TF-IDF scores we used Gensim, a Python and NumPy package for vector space modelling of text documents. These features were collected in a vector space model representing the data set.

IV. Base Classifiers and Ensemble Classifiers

The following classifiers available in WEKA [13] were used as base classifiers in our experiments with outcomes presented in Section V: FURIA [17], J48 [37], LibLINEAR [9], LibSVM [7], [10], [16], Random Forest [6], SMO [15], [27], [36], These robust classifiers were chosen since they represent most essential types of classifiers available in WEKA [13] and performed well for our data set.

We used SimpleCLI command line in WEKA [13] to investigate the performance of the following ensemble techniques: AdaBoost [12], Bagging [5], Dagging [39], Decorate [32], Grading [38], MultiBoost [43] and Stacking [46].

Consensus functions can also be used as a replacement for voting to combine the outputs of several classifiers. Here we use the HBGF consensus function, following the recommendations of [11] and our previous experience with consensus functions presented in [8], [47] and [48]. The HBGF consensus function is based on a bipartite graph with two sets of vertices: classes and elements of the data set.

V. Experiments Evaluating Performance

We used 10-fold cross validation to evaluate the effectiveness of classifiers in all experiments. The following measures of performance of classifiers are often used in this research direction: precision, recall, F-measure, accuracy, sensitivity, specificity and Area Under Curve also known as the Receiver Operating Characteristic or ROC area.

Notice that weighted average values of the performance metrics are usually used. This means that they are calculated for each class separately, and a weighted average is found then. In particular, our results included in this paper deal with the weighted average values of precision. In contrast, the *accuracy* is defined for the whole classifier as the percentage of all websites classified correctly, which means that this definition does not involve weighted averages in the calculation. *Precision* of a classifier, for a given class, is the ratio of true positives to combined true and false positives.

Sensitivity is the proportion of positives (phishing websites) that are identified correctly. *Specificity* is the proportion of negatives (legitimate websites) which are identified correctly. Sensitivity and specificity are measures evaluating binary classifications. For multi-class classifications they can be also used with respect to one class and its complement. Sensitivity is also called True Positive Rate. *False Positive Rate* is equal to 1 - specificity. These measures are related to recall and precision. *Recall* is the ratio of true positives to the number of all positive samples (i.e., to the combined true positives and false negatives). The recall calculated for the class of phishing websites is equal to sensitivity of the whole classifier.

All tables of outcomes in this paper include the F-measure, since it combines precision and recall into a single number evaluating performance of the whole system, [44]. The F-measure is equal to the harmonic mean of precision and recall

$$\text{F-measure} = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} \tag{1}$$

The weighted average F-measure is contained in the standard WEKA output for all classifiers.

First, we include the results of experiments comparing the performance of several base classifiers for phishing websites. The results obtained for five best classifiers are presented in Figure 3. Random Forest outperformed other base classifiers for the phishing websites data set.

Second, we include the results of experiments comparing standard ensemble classifiers in their ability to improve the outcomes. We compared AdaBoost, Bagging, Dagging, Decorate, Grading, HBGF, MultiBoost and Stacking based on RandomForest.

F-measures of the resulting ensemble classifiers are presented in Figure 4, which shows improvement as compared to the base classifiers. In these tests all ensembles were used

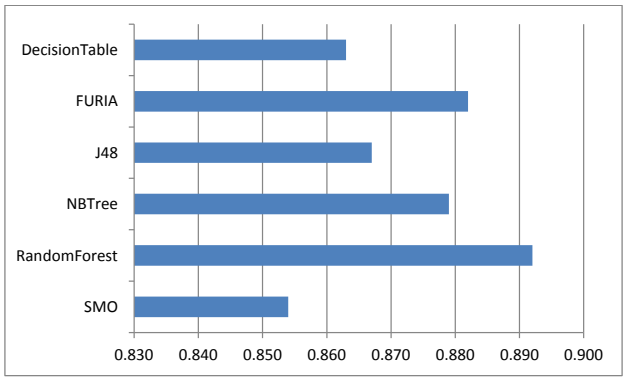

Figure 3. F-measure of base classifiers for phishing websites

with one and the same base classifier, RandomForest, in all tests.

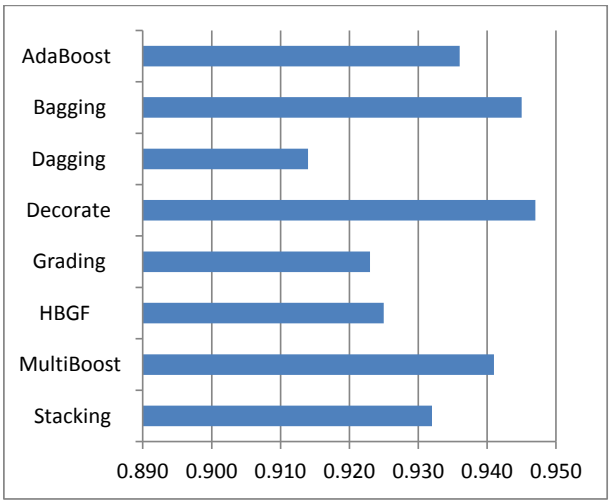

Figure 4. F-measure of ensemble classifiers for phishing websites

Finally, we include the results of experiments evaluating the 3-tier ensemble method. This is the main topic of the paper. These experiments included the all combinations of Bagging, Decorate and MultiBoost, since these ensemble methods produced better F-measures in Figure 4. Each three-tier ensemble classifier contains one ensemble in Tier 3. It generates or includes a whole set of Tier 2 ensembles and executes them in exactly the same way as it handles any other base classifiers. In turn, each Tier 2 ensemble applies its method to combine its base classifiers in Tier 1. We have not included repetitions of the same ensemble technique in both tiers, since tests have shown that they do not produce further improvement. The outcomes of the three-tier ensemble classifiers are presented in Figure 5. A part of command generating one of these multi-level ensembles in SimpleCLI is shown in Figure 2.

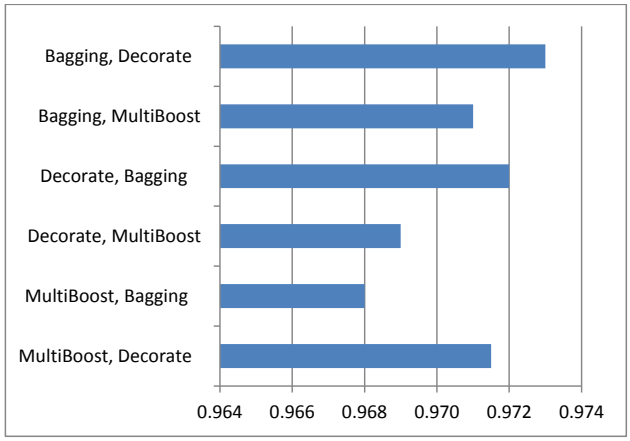

Figure 5. F-measure of three-tier ensemble classifiers for phishing websites

VI. Discussion

Our work shows that large three-tier ensamples are quite easy to use and can be applied to improve classifications, if diverse ensembles are combined at different tiers. It is an interesting question for future research to investigate three-tier ensembles for other large datasets.

Random Forest outperformed other base classifiers for the phishing websites data set, and Decorate improved its outcomes better than other ensemble meta classifiers did. The best outcomes were obtained by the new combined three-tier ensemble classifier where Bagging is used in Tier 3 and Decorate in Tier 2.

The performance of ensemble classifiers considered in this paper depends on several numerical input parameters. In all experiments we used them with the same default values of these parameters in order to have a uniform equivalent comparison of outcomes across all of these ensemble classifiers. It may be also possible to obtain further improvement to the outcomes by optimizing their parameters with optimization techniques presented in [3]. At present the ranges of parameter values remain restricted by the size of memory available on personal computers for training of large three-tier ensemble classifiers.

VII. Conclusion

We carried out a systematic investigation of new automatically generated multi-tier ensemble classifiers, where diverse ensembles are combined into a unified system by integrating different ensembles at a lower tier as a part of another ensemble at the top tier. Our experiments evaluated the performance of these large three-tier ensemble classifiers for a data set of phishing websites and have demonstrated the

feasibility and performance of the approach. The experimental outcomes show that these multi-tier ensemble classifiers can be used to improve classifications. They produced better results compared to the base classifiers or standard ensemble classifiers.

ACKNOWLEDGMENT

The authors are grateful to four referees for thorough reports with comments and corrections that have helped to improve the text of this article, and for suggesting several possible directions for future research. All authors were supported by Deakin-Ballarat collaboration grants.

REFERENCES

[1] APWG, "Anti-Phishing Working Group," http://apwg.org/, accessed 10 June 2012.

[2] L. Batten, J. Abawajy, and R. Dose, "Prevention of information harvesting in cloud service environments," in *Proceedings of the 1st International Conference on Cloud Computing and Services Science, CLOSER 2011*, 2011, pp. 66–72.

[3] G. Beliakov and J. Ugon, "Implementation of novel methods of global and non-smooth optimization: GANSO programming library," *Optimization*, vol. 56, pp. 543–546, 2007.

[4] G. Beliakov, J. Yearwood, and A. Kelarev, "Application of rank correlation, clustering and classification in information security," *Journal of Networks*, vol. 7, pp. 935–955, 2012.

[5] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, pp. 123–140, 1996.

[6] ——, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

[7] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[8] R. Dazeley, J. Yearwood, B. Kang, and A. Kelarev, "Consensus clustering and supervised classification for profiling phishing emails in internet commerce security," in *Knowledge Management and Acquisition for Smart Systems and Services, PKAW2010*, ser. Lecture Notes in Computer Science, vol. 6232, 2010, pp. 235–246.

[9] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR – a library for large linear classification," Software available at http://www.csie.ntu.edu.tw/~cjlin/liblinear/, viewed 21 February 2012, 2012.

[10] R.-E. Fan, P.-H. Chen, and C.-J. Lin, "Working set selection using second order information for training SVM," *J. Machine Learning Research*, vol. 6, pp. 1889–1918, 2005.

[11] X. Fern and C. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning," in *21st International Conference on Machine Learning, ICML'04*, vol. 69. New York, NY, USA: ACM, 2004, pp. 36–43.

[12] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *Proc. 13th Internat. Conf. Machine Learning*, 1996, pp. 148–156.

[13] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten, "The WEKA data mining software: an update," *SIGKDD Explorations*, vol. 11, pp. 10–18, 2009.

[14] I. Hamid and J. Abawajy, "Hybrid feature selection for phishing email detection," in *International Conference on Algorithms and Architectures for Parallel Processing, ICA3PP 2011*, ser. Lecture Notes in Computer Science, vol. 7017, 2011, pp. 266–275.

[15] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," in *Advances in Neural Information Processing Systems*, 1998.

[16] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," Dept. Computer Science, National Taiwan University, http://www.csie.ntu.edu.tw/~cjlin, Initial version: 2003, last updated: April 15, 2010.

[17] J. Huehn and E. Huellermeier, "FURIA: An algorithm for unordered fuzzy rule induction," *Data Mining and Knowledge Discovery*, vol. 19, pp. 293–319, 2009.

[18] R. Islam and J. Abawajy, "A multi-tier phishing detection and filtering approach," *Journal of Network and Computer Applications*, p. to apper soon, 2012.

[19] R. Islam, J. Abawajy, and M. Warren, "Multi-tier phishing email classification with an impact of classifier rescheduling," in *10th International Symposium on Pervasive Systems, Algorithms, and Networks, ISPAN 2009*, 2009, pp. 789–793.

[20] R. Islam, J. Singh, A. Chonka, and W. Zhou, "Multi-classifier classification of spam email on an ubiquitous multi-core architecture," in *Proceedings – 2008 IFIP International Conference on Network and Parallel Computing, NPC 2008*, 2008, pp. 210–217.

[21] R. Islam, R. Tian, L. Batten, and S. Versteeg, "Classification of malware based on string and function feature selection," in *CTC 2010: Proceedings of the Second Cybercrime and Trustworthy Computing Workshop*, 2010, pp. 9–17.

[22] R. Islam, R. Tian, V. Moonsamy, and L. Batten, "A comparison of the classification of disparate malware collected in different time periods," *Journal of Networks*, vol. 7, pp. 956–955, 2012.

[23] R. Islam and W. Zhou, "Email classification using multi-tier classification algorithms," in *Proc. 7th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2008*, 2008.

[24] R. Islam, W. Zhou, and M. Chowdhury, "Email categorization using (2+1)-tier classification algorithms," in *Proceedings – 7th IEEE/ACIS International Conference on Computer and Information Science, IEEE/ACIS ICIS 2008, In conjunction with 2nd IEEE/ACIS Int. Workshop on e-Activity, IEEE/ACIS IWEA 2008*, 2008, pp. 276–281.

[25] R. Islam, W. Zhou, M. Gao, and Y. Xiang, "An innovative analyser for multi-classifier email classification based on grey list analysis," *Journal of Network and Computer Applications*, vol. 32, pp. 357–366, 2009.

[26] B. Kang, A. Kelarev, A. Sale, and R. Williams, "A new model for classifying DNA code inspired by neural networks and FSA," in *Advances in Knowledge Acquisition and Management*, ser. Lecture Notes in Computer Science, vol. 4303, 2006, pp. 187–198.

[27] S. Keerthi, S. Shevade, C. Bhattacharyya, and K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," *Neural Computation*, vol. 13, no. 3, pp. 637–649, 2001.

[28] A. Kelarev, S. Brown, P. Watters, X.-W. Wu, and R. Dazeley, "Establishing reasoning communities of security experts for internet commerce security," in *Technologies for Supporting Reasoning Communities and Collaborative Decision Making: Cooperative Approaches*. IGI Global, 2011, pp. 380–396.

[29] A. Kelarev, B. Kang, and D. Steane, "Clustering algorithms for ITS sequence data with alignment metrics," in *AI 2006: Advances in Artificial Intelligence, 19th Australian Joint Conference on Artificial Intelligence*, ser. Lecture Notes in Artificial Intelligence, vol. 4304, 2006, pp. 1027–1031.

[30] A. Kelarev, R. Dazeley, A. Stranieri, J. Yearwood, and H. Jelinek, "Detection of CAN by ensemble classifiers based on Ripple Down Rules," in *Pacific Rim Knowledge Acquisition Workshop, PKAW2012*, ser. Lecture Notes in Artificial Intelligence, vol. 7457, 2012, pp. 147–159.

[31] A. Kelarev, A. Stranieri, J. Yearwood, and H. Jelinek, "Empirical study of decision trees and ensemble classifiers for monitoring of diabetes patients in pervasive healthcare," in *Network-Based Information Systems, NBiS-2012*, 2012, pp. 441–446.

[32] P. Melville and R. Mooney, "Creating diversity in ensembles using artificial data," *Information Fusion*, vol. 6, pp. 99–111, 2005.

[33] V. Moonsamy, R. Tian, and L. Batten, "Feature reduction to speed up malware classification," in *Information Security Technology for Applications*, ser. Lecture Notes in Computer Science, P. Laud, Ed. Springer Berlin / Heidelberg, 2012, vol. 7161, pp. 176–188.

[34] OECD, "Organisation for Economic Cooperation and Development, OECD task force on spam, OECD anti-spam toolkit and its annexes," http://www.oecd.org/dataoecd/63/28/36494147.pdf, accessed 20 November 2011.

[35] PhishTank, "Developer information," http://www.phishtank.com/developer_info.php, viewed 20 September 2011.

[36] J. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods – Support Vector Learning*, 1998.

[37] R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann, 1993.

[38] A. Seewald and J. Fuernkranz, "An evaluation of grading classifiers advances in intelligent data analysis," in *Advances in Intelligent Data Analysis*, ser. Lecture Notes in Computer Science, vol. 2189/2001, 2001, pp. 115–124.

[39] K. Ting and I. Witten, "Stacking bagged and dagged models," in *Fourteenth international Conference on Machine Learning*, 1997, pp. 367–375.

[40] H. Vu, G. Li, and G. Beliakov, "A fuzzy decision support method for customer preferences analysis based on Choquet Integral," in *IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2012*, 2012, pp. 1–8.

[41] H. Vu, S. Liu, Z. Li, and G. Li, "Microphone identification using one-class classification approach," in *Applications and Techniques in Information Security, ATIS 2011*, 2011, pp. 29–37.

[42] X. Wang, W. Niu, G. Li, X. Yang, and Z. Shi, "Mining frequent agent action patterns for effective multi-agent-based web service composition," in *7th International Workshop on Agents and Data Mining Interation, ADMI 2011*, ser. Lecture Notes in Artificial Intelligence, vol. 7103, 2012, pp. 211–227.

[43] G. Webb, "Multiboosting: A technique for combining boosting and wagging," *Machine Learning*, vol. 40, pp. 159 – 196, 2000.

[44] I. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. Amsterdam: Elsevier/Morgan Kaufman, 2011.

[45] D. Wolpert, "The lack of a priori distinctions between learning algorithms," *Neural Computation*, vol. 8, pp. 1341–1390, 1996.

[46] ——, "Stacked generalization," *Neural Networks*, vol. 5, pp. 241–259, 1992.

[47] J. Yearwood, D. Webb, L. Ma, P. Vamplew, B. Ofoghi, and A. Kelarev, "Applying clustering and ensemble clustering approaches to phishing profiling," in *Data Mining and Analytics 2009, Proc. 8th Australasian Data Mining Conference, AusDM 2009*, ser. CRPIT, P. Kennedy, K. Ong, and P. Christen, Eds., vol. 101. Melbourne, Australia: ACS, 2009, pp. 25–34.

[48] J. Yearwood, B. Kang, and A. Kelarev, "Experimental investigation of classification algorithms for ITS dataset," in *Pacific Rim Knowledge Acquisition Workshop, PKAW 2008*, Hanoi, Vietnam, 15–16 December 2008, 2008, pp. 262–272.